AP/2176
4 IDW

Attorney's Docket No.: 07844-273001 / P247

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Michaud et al.
Serial No. : 09/058,496
Filed : April 10, 1998
Title : ASSIGNING A HOT SPOT IN AN ELECTRONIC ARTWORK

Art Unit : 2176
Examiner : William Bashore

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

SUBMISSION OF APPEAL BRIEF

Further to the Notice of Appeal filed on October 19, 2004, and received in the U.S. Patent Office on October 22, 2004, the Applicant submits herewith an Appeal Brief, and a check in the amount of $500.00 for the Appeal Brief fee.

Respectfully submitted,

Date: December 21, 2004

Jennifer A. Zanocco
Reg. No. 54,563

Customer No. 021876
Fish & Richardson P.C.
Telephone: (650) 839-5070
Facsimile: (650) 839-5071

50252925.doc

CERTIFICATE OF MAILING BY FIRST CLASS MAIL

I hereby certify under 37 CFR §1.8(a) that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

December 21, 2004
Date of Deposit

Carlos A. Brasil
Signature

Carlos A. Brasil
Typed or Printed Name of Person Signing Certificate

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Michaud et al. Art Unit : 2176
Serial No. : 09/058,496 Examiner : William Bashore
Filed : April 10, 1998
Title : ASSIGNING A HOT SPOT IN AN ELECTRONIC ARTWORK

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

(1) Real Party in Interest

The real party in interest is Adobe Systems Incorporated.

(2) Related Appeals and Interferences

None.

(3) Status of Claims

Claims 1-9, 12-13, 15-16, 20-22, 24-25 and 28-44 are pending.

Claims 1-9, 12-13, 15-16, 20-22, 24-25, 28-36 and 38-43 are rejected.

Claims 37 and 44 are objected to.

All of the foregoing claims are being appealed.

(4) Status of Amendments

Claims 1 and 3 were amended in a Supplemental Amendment filed December 20, 2004. No other pending claims include unentered amendments.

CERTIFICATE OF MAILING BY FIRST CLASS MAIL

I hereby certify under 37 CFR §1.8(a) that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Dec. 21, 2004
Date of Deposit

Carla C. Grant
Signature

Carlos A. Brasic
Typed or Printed Name of Person Signing Certificate

(5) Summary of Claimed Subject Matter

An area, or "hot spot", and a corresponding action are associated with a selected layer of an electronic artwork (Specification, page 1, lines 8-10). The electronic artwork has multiple layers composited together (Specification, page 4, lines 18-19). Each layer typically includes image data and optionally a mask and compositing controls (Specification, page 4, lines 25-26). The image data can be represented by an array of pixels, with each pixel having a color and an opacity (Specification, page 4, lines 26-28).

A user selects a layer from the electronic artwork (Specification, page 5, lines 17-22). The layer has one or more non-transparent regions (Specification, page 7, lines 1-3). A perimeter boundary is calculated (Specification, page 7, lines 5-8). The perimeter boundary is used to define an area in the selected layer (*id.*). For example, the outer boundary can be traced to create a polygon approximating the outer boundary of the region's pixels (Specification, page 7, lines 11-14). The polygon defines the "hot spot" (Specification, page 7, lines 15-18).

An action is assigned to the area, the action defining a function that is to be activated when the area is selected (Specification, page 6, lines 4-6). For example, the user can indicate an action to be associated with the hot spot by inputting a URL (*id.*). When the action is assigned to a layer, the area and the action are associated with the layer as a property of the layer (Specification, page 6, lines 10-15).

Independent claim 5 is directed to a computer program that performs the method of claim 1.

Claim 3 depends from claim 1 and includes compositing the plurality of layers by combining the plurality of layers in accordance to produce a final image and converting the area and the action to a target output format (Specification, page 6, lines 16-28).

Claim 6 depends from claim 5 and recites instructions to automatically fit a shape to the perimeter boundary, wherein the shape defines the area (Specification, page 7, lines 5-7).

Claim 12 depends from claim 1 and recites re-defining the area automatically if the content of the selected layer of the electronic artwork is edited, the re-defined area conforming to a new perimeter boundary of the one or more non-transparent regions (Specification, page 3, lines 14-16).

Claim 13 depends from claim 3, which depends from claim 1, and recites calculating any dynamic content for the selected layer when the layer is composited and using the calculated dynamic content to calculate the perimeter boundary and define the area (Specification, page 6, lines 28-31).

Claim 15 depends from claim 1 and recites that the image data in the selected layer has two or more non-contiguous non-transparent regions and the two or more non-contiguous non-transparent regions in combination are used to calculate the perimeter boundary (Specification, page 7, lines 1-5).

Claim 35 depends from claim 1 and recites that the perimeter boundary is for one or more non-transparent regions in combination (Specification, page 7, lines 15-18).

Claim 36 depends from claim 1 and recites that there are one or more holes within the one or more non-transparent regions and the holes are ignored in calculating the perimeter boundary (Specification, page 8, lines 7-9).

Claim 37 depends from claim 1 and recites that wherein there are one or more holes within the one or more non-transparent regions that each non-transparent region that has a hole is separated into separate non-transparent regions that do not contain holes and the perimeter boundary is calculated from the separate hole free non-transparent regions (Specification, page 7, lines 8-12).

(6) Grounds of Rejection

Claims 1-5, 7-9, 12-13, 20-22, 29-30, 35-36 and 42-43 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Mapedit Imagemap Editing Software, Version 2.3 for Windows 3.1, Boutell.Com, Inc. (1997), *at* http://www.boutell.com/mapedit ("Mapedit"), in view of U.S. Patent No. 6,034,689 ("White").

Claims 15-16 and 24-25 stand rejected under 35 U.S.C. § 103 (a) as being unpatentable over Mapedit and White as applied to claims 1 and 5, and further in view of U.S. Patent No. 5,999,781 ("Nielsen").

Claims 6, 28, 31-34 and 38-41 stand rejected under 35 U.S.C. § 103 (a) as being unpatentable over Mapedit and White as applied to claim 1, and further in view of U.S. Patent No. 5,956,701 ("Habermehl").

Claims 37 and 44 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form, including all of the limitations of the base claim and any intervening claims.

(7) Argument

I. Claims 1-5, 7-9, 12-13, 20-22, 29-30, 35-36 and 42-43 are not properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Mapedit in view of White.

Claims 1-5, 7-9, 12-13, 20-22, 29-30, 35-36 and 42-43

Claim 1 recites a method. The method includes receiving from a user an input selecting a layer from among multiple layers in an electronic artwork, each layer includes image data, and the image data of the selected layer includes opacity data. The opacity data is used to identify one or more non-transparent regions. A perimeter boundary of the non-transparent regions is calculated and is used to define an area in the selected layer. An action is assigned to the area, the action defining a function that is to be activated when the area is selected. The area and the action are associated with the selected layer as a property of the selected layer in the electronic artwork.

Neither Mapedit nor White, nor the combination of them, disclose or teach the claimed subject matter.

Mapedit and White do not teach or disclose "receiving from a user an input selecting a layer in an electronic artwork having a plurality of layers, each layer including image data, the image data of the selected layer including opacity data;" and "using the opacity data of the selected layer to identify one or more non-transparent regions."

The above recites the first two limitations of claim 1. The primary reference relied on by the Examiner is Mapedit. Mapedit is a WYSIWYG (What You See Is What You Get) editor for image map files. Mapedit describes loading a pre-existing image in GIF, JPEG or PNG into a scrollable, resizeable window (Fig. 4, paragraph 9). A user can create an image map using the image by manually drawing polygons or circles on top of the image and specifying a URL for each polygon or circle (Fig. 8, paragraph 4). The polygon or circle in combination with the URL creates a hotspot (*id.*). The user can edit the hotspot color (Fig. 11, paragraph 1). When a

viewer of the image clicks on the hotspot, the region within the hotspot will be displayed in reverse video (Fig. 12, paragraph 1, Figs. 17 and 18).

Neither Mapedit nor White, alone or in combination, teach or disclose multiple layers having image data, at least one of which (*i.e.*, the selected layer) has opacity data. Layers can be thought of as stacked sheets of acetate with relative positions above and below one another (Specification, page 4, lines 18-24). The density of ink on each acetate sheet can be represented as the opacity or alpha value (*id.*). Thus, the opacity data of a given layer prescribes whether and how much of a layer below the given layer appears in a final composited image. Mapedit describes drawing polygons over an image to create a hotspot (Figure 10, paragraph 1). The user subsequently can edit the hotspot by selecting a color for the hotspot (Figure 11, paragraph 1). However, the Mapedit does not suggest or disclose an opacity for the circles and hotspots.

The Examiner asserts that selecting an overlapping image results in a translucent image (Non-Final Office Action, page 3). In particular, the Examiner asserts that Mapedit's selection of an overlapping polygon causes the polygon to become translucent and that translucency is a degree of opaqueness, therefore the selection involves opacity data so that the translucent image can be displayed (Non-Final Office Action, page 9). The Examiner misunderstands the meaning of a layer including image data that includes opacity data. What Mapedit teaches is reverse video (Figure 12, paragraph 1), or "a display method that causes a portion of the display to appear like a negative of the regular display. If the display screen normally displays light images against a dark background, putting it in reverse video mode will cause it to display dark images against a light background" (Webopedia, www.webopedia.com). Reverse video changes the display. Even accepting the Examiner's assertions for the sake of argument, merely displaying elements having different opacity does not satisfy the limitation of "using opacity data of the selected layer to identify one or more non-transparent regions."

There is no opacity data disclosed by Mapedit and therefore no opacity data is used to identify non-transparent regions. Further, Mapedit teaches that a user creates the hotspot by drawing a circle or a polygon on top of an image. The hotspots are not identified by opacity data, but by the outline of the circle or polygon created by the user.

Mapedit and White do not teach or disclose "calculating a perimeter boundary of the one or more non-transparent regions."

White teaches a scaling operation that reduces an image size of all web page elements to fit within the horizontal dimensions of a television-formatted display (col. 15, lines 1-3). The scaling operation is performed on both the horizontal and vertical dimensions using the same scale factor (col. 15, lines 3-6). The Examiner states "It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the method of White to the method of Mapedit, because of White's taught advantage of scaling, providing a way for Mapedit to create imagemaps for different presentation mediums by automatically defining image map boundaries subsequent to changes in size of an imagemap" (Non-Final Office Action, page 4).

Applicant respectfully submits the Examiner has misconstrued applicant's claim 1. Applicant's claim 1 recites a method whereby a user selects a layer that includes image data and opacity data. The opacity data of the selected layer is used to identify one or more non-transparent regions. A perimeter boundary of the non-transparent regions is calculated. The perimeter boundary defines an area in the selected layer.

Scaling all web page elements so that both the horizontal and vertical dimensions are scaled using the same factor causes all of the elements on a web page to be enlarged or reduced according to a particular value. A scaling factor is determined based on the dimensions of the display, and rescaled X and Y coordinates are determined. Scaling does not calculate a perimeter boundary of non-transparent regions. Further, White does not identify non-transparent regions and does not use opacity data to identify non-transparent regions.

Claim 1 requires defining an area by calculating a perimeter boundary of a non-transparent region, *i.e.*, a perimeter boundary of a non-transparent region as it existed in the layer when the layer was selected by the user. It is the act of calculating a perimeter boundary of a non-transparent region that defines the area. This is different from modifying a perimeter boundary by rescaling, which modified perimeter boundary then encloses and defines an area.

To establish a *prima facie* case of obviousness, the Examiner must make three basic showings. First, there must be some suggestion or motivation, either in the references or in the prior knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success.

Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on Applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Mapedit and White both fail to disclose or suggest receiving from a user an input selecting a layer in an electronic artwork having a plurality of layers, each layer including image data, the image data of the selected layer including opacity data; using the opacity data of the selected layer to identify one or more non-transparent regions or calculating a perimeter boundary of the one or more non-transparent regions, as required by claim 1. Thus, a *prima facie* case of obviousness under 35 U.S.C. § 103 has not been established for claim 1. Claim 5 is an analogous computer product claim to claim 1. Claims 2-4, 12-13, 29 and 35-36 depend directly or indirectly from claim 1 and claims 7-9, 20-22, 30 and 42-43 depend directly or indirectly from claim 5. Accordingly, claims 1-5, 7-9, 12-13, 20-22, 29-30, 35-36 and 42-43 are allowable.

Claims 3 and 7

Mapedit and White do not teach or disclose compositing controls.

Amended claim 3 depends from claim 1 and recites a method where the layers include compositing controls. Layers are combined to build a final image. Claim 7 is an analogous computer program claim to claim 3.

Mapedit in view of White also does not disclose compositing the layers of an image by combining the plurality of layers to build a final image, as required by claim 3. The Examiner asserts that Mapedit teaches the compositing of images (Non-Final Office Action, page 5). The Examiner notes that Mapedit teaches a displayed image made up of a main image and mapped polygon areas, which work as a final combined group, which can be interpreted as a final composited image (Non-Final Office Action, pages 9 and 10).

Compositing is a process of combining multiple images into a single image (Specification, page 4; Edinburgh Online Graphics Dictionary, http://homepages.inf.ed.ac.uk/rbf/GRDICT/grdict.htm). When layers including opacity data are composited, the opacity data is used to calculate the colors in the final image. For example, a layer with an area having an opacity less than 100% will allow corresponding areas in layers

below to show through, whereas if the same area has an opacity of 100%, the areas below would be obscured. Compositing controls determine how colors in a layer combine with colors from other layers. Mapedit writes a map, made up of the polygons, circles and URLs associated with the polygons and circles, to the image (Figure 19). Creating a map file for an image does not teach or disclose compositing layers of an image by combining the plurality of layers to build a final image.

A layer with a compositing control can affect the appearance of the layer or of other layers in the stack, thereby affecting the appearance of the composited final image. Compositing layers having opacity data and compositing controls produces a different final image than the final image produced when layers without opacity data and compositing controls are stacked upon one another. What Mapedit teaches is drawing a polygon on top of an image. Mapedit does not teach or disclose compositing layers to build a final image where the plurality of layers include compositing controls.

Mapedit and White in combination fail to disclose compositing controls as recited in claims 3 and 7, and a *prima facie* case of obviousness has not been established. The applicant submits claims 3 and 7 are not properly rejected.

Claims 12 and 21

Mapedit and White do not teach or disclose re-defining an area automatically if the content of the selected layer of the electronic artwork is edited.

Claim 12 recites the additional feature of re-defining the area automatically if the content of the selected layer of the electronic artwork is edited, to conform to a new perimeter boundary of the one or more non-transparent regions. Claim 21 is an analogous computer program claim to claim 12.

Again, the Examiner contends that White teaches the rescaling of an image map area subsequent to the resizing of a web page to fit different display areas (Non-Final Office Action, page 6). However, White merely discloses rescaling an image map area when a web page is resized to fit different display areas, not changing the hotspot area to conform to content of a layer when the underlying image is edited.

Because Mapedit and White fail to disclose the elements of claims 12 and 21, a *prima facie* case of obviousness has not been established. The applicant submits claims 12 and 21 are not properly rejected.

Claims 13 and 22

Mapedit and White do not teach or disclose calculating any dynamic content for the selected layer before the area is defined.

Claim 13 is also patentable because it recites the additional feature of calculating any dynamic content for the selected layer before the area is defined. Claim 22 is an analogous computer program claim to claim 13.

The specification defines "dynamic content" as data that is computed from other data at the time the layers are composited (Specification, page 5, lines 3-5). The Examiner states, "Since it is known in the art that currently edited information is considered dynamic information until saved, Mapedit's calculation and formulation of hotspots is based upon dynamic content, prior to saving" (Non-Final Office Action, page 5). Even assuming that "currently edited information" can be considered dynamic, it is not immediately apparent how Mapedit's manual definition of hotspot circles and polygons discloses "calculating dynamic content for the selected layer before the area is defined" as claim 13 requires. But more importantly, the rejection appears to ignore that "dynamic content" is given a specific definition in the specification - that is, content that is computed from other data at the time the layers are composited. Mapedit merely discloses a method of adding hotspots to a static image and does not appear to even suggest any way to associate a hotspot with an image layer having dynamic content as that term is defined in the specification.

Because Mapedit and White fail to disclose the elements of claims 13 and 22, a *prima facie* case of obviousness has not been established. The applicant submits claims 13 and 22 are not properly rejected.

Claims 35 and 42

Mapedit and White do not teach or disclose that the perimeter boundary is for the one or more non-transparent regions in combination.

Claim 35 recites that the perimeter boundary is for the one or more non-transparent regions in combination. Claim 42 is an analogous computer program claim to claim 35.

The Examiner states, "Mapedit teaches determination of a perimeter of a non-transparent region" (Non-Final Office Action, page 7). As discussed above in related claim 1, Mapedit does not disclose calculating a perimeter boundary, but requires a user to manually draw a circle or polygon, and to specify a URL for each such shape on the image (Figure 8). Even if a manually drawn shape is a perimeter of a non-transparent region (which it is not), in Mapedit, if one hotspot overlaps another, "the oldest gets the click" (Figure 12). This indicates that the polygons are not in combination, but rather each polygon remains a separate hotspot. Accordingly, Mapedit fails to disclose calculating a perimeter boundary for one or more non-transparent regions in combination.

Because Mapedit and White fail to disclose the elements of claims 35 and 42, a *prima facie* case of obviousness has not been established. The applicant submits claims 35 and 42 are not properly rejected.

Claims 36 and 43

Mapedit and White do not teach or disclose that holes included within the perimeter boundary are included in the area.

Claim 36 recites that the content of the selected layer includes one or more holes formed between the one or more non-transparent regions and the holes included within the perimeter boundary are included in the area. Claim 43 is an analogous computer program claim 36.

The Examiner turns to Mapedit as teaching creating and defining a region as a superset of another region (Non-Final Office Action, page 7). However, the applicant respectfully suggests the Examiner has misconstrued Mapedit and claim 36. Mapedit shows a first circular region drawn on an image with no associated hyperlink and a second circular region with an associated hyperlink overlapping the first circular image (Figures 20-23). Due to the "the oldest gets the click" (Figure 12) scheme, the first region is not necessarily associated with the second region's hyperlink and the first region remains separate from the second region. Although the two regions overlap, they retain independent actions. By contrast, the applicant's invention as recited in claim 36 requires that holes included within the perimeter boundary are included in the area, *i.e.*, an action will be assigned to the hole, so there are not non-hot spots within an area. Mapedit does not disclose holes formed between one or more non-transparent regions and including holes that are within the perimeter boundary in the area. Mapedit shows that the first circular region is

not associated with a hyperlink, while the second circular region, which surrounds and overlaps the first circular region, is associated with a hyperlink. The first circular region is excluded from activating the second circular region's hyperlink.

Because Mapedit and White in combination fail to disclose the elements of claims 36 and 43, a *prima facie* case of obviousness has not been established. The applicant submits claims 36 and 43 are not properly rejected.

II. Claims 15-16 and 24-25 are not properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Mapedit and White and further in view of Nielsen.

Mapedit, White and Nielsen do not teach or disclose that an area is defined by automatically determining a perimeter boundary of two or more non-transparent regions in combination.

Claim 15 depends from claim 1 and further defines the selected layer as having two or more non-contiguous non-transparent regions in a transparent frame and recites that the area is defined by automatically determining a perimeter boundary of the non-transparent regions in combination. That is, the hotspot includes the non-contiguous non-transparent regions in combination. Claim 16 recites a method that further comprises generating multiple image maps from the non-transparent regions. Claims 24 and 25 are analogous computer program claims.

The applicant respectfully submits the Examiner has misconstrued Nielsen as disclosing the elements of claim 15. Nielsen discloses having more than one contiguous region and having a different action associated with each contiguous region (Fig. 1b, col. 2, lines 46-53).

Mapedit, White and Nielson fail to disclose the elements of claim 15, and a *prima facie* case of obviousness has not been established. Claim 16 depends from claim 15 and claims 24-25 are analogous computer program claims to claims 15 and 16 and are similarly not properly rejected for at least the above reasons.

III. Claims 6, 28, 31-34 and 38-41 are not properly rejected under 35 U.S.C. § 103 (a) as being unpatentable over Mapedit and White and further in view of Habermehl.

Mapedit, White and Habermehl do not teach automatically fitting a shape to a perimeter boundary.

Claim 6 recites a computer program claim, dependent on claim 5, that further comprises instructions to automatically fit a shape to the perimeter boundary, wherein the shape defines the area. Claim 28 is an analogous method claim.

Habermehl describes a method where "the user defines the specified region by selecting points within the region by performing an act such as randomly clicking an input device such as a mouse, associated with a cursor, within the specified region" (col. 3, lines 22-25). Habermehl does not teach automatically fitting a shape to the perimeter boundary. Habermehl requires the user to click at least three times to define an area within the region (col. 3, lines 25-29). "[A] boundary which is more complex will require a larger number of clicks than a boundary which is relatively simple" (col. 3, lines 34-36). The Examiner fails to explain how Habermehl discloses automatically fitting a shape to a perimeter boundary.

The applicant submits Mapedit, White and Habermehl fail to disclose the elements of claim 6, and no *prima facie* case of obviousness has been made. Claims 31-34 depend from claim 28, claims 38-41 depend from claim 6. For at least the reasons stated above, these claims are similarly not properly rejected.

IV. Claims 37 and 44 are not properly objected to.

Claim 37 depends from claim 1 and claim 44 depends from claim 5. As shown above, claims 1 and 5 are not properly objected to over the combination of Mapedit and White. Therefore, claims 37 and 44 need not be rewritten in independent form, including all of the limitations of the base claim and any intervening claims and are allowable as they stand.

The brief fee of $500 is enclosed. Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: December 21, 2004

Jennifer A. Zanocco
Reg. No. 54,563

Customer No. 021876
Fish & Richardson P.C.
Telephone: (650) 839-5070
Facsimile: (650) 839-5071

50250130.doc

Claims Appendix

1. (Previously Presented) A method comprising:

receiving from a user an input selecting a layer in an electronic artwork having a plurality of layers, each layer including image data, the image data of the selected layer including opacity data;

using the opacity data of the selected layer to identify one or more non-transparent regions;

calculating a perimeter boundary of the one or more non-transparent regions;

using the perimeter boundary to define an area in the selected layer;

assigning an action to the area, the action defining a function that is to be activated when the area is selected; and

associating the area and the action with the selected layer as a property of the selected layer in the electronic artwork.

2. (Original) The method of claim 1, wherein:

the action is a URL (Uniform Resource Locator).

3. (Previously Presented) The method of claim 1, wherein the plurality of layers include compositing controls and the method further comprises:

compositing the plurality of layers of the artwork by combining the plurality of layers to produce a final image; and

converting the area and the action to a target output format.

4. (Original) The method of claim 3, wherein:

the target output format is HTML (HyperText Markup Language).

5. (Previously Presented) A computer program, tangibly stored on a computer-readable medium, comprising instructions for causing a computer to:

receive an electronic artwork having a plurality of layers, each layer including image

data;

receive from a user an input selecting one of the plurality of layers, the image data of the selected layer including opacity data;

use the opacity data to identify one or more non-transparent regions in the image data;

calculate a perimeter boundary of the one or more non-transparent regions;

use the perimeter boundary to define an area in the selected layer; and

assign an action to the area, the action defining a function to be activated when the area is selected.

6. (Previously Presented) The computer program of claim 5, further comprising instructions to:

automatically fit a shape to the perimeter boundary, wherein the shape defines the area.

7. (Previously Presented) The computer program of claim 5, wherein the plurality of layers include compositing controls and the program further comprises instructions to:

composite the artwork by combining the plurality of layers to produce a final image; and

convert the area and the action to a target output format.

8. (Original) The computer program of claim 7, wherein the target output format for the area and the action is HTML.

9. (Previously Presented) The computer program of claim 8, further comprising instructions to:

write out the final image as an image file and write out an HTML file containing an image map for the area and a URL for the action, the HTML file referring to the image file.

10-11. (Cancelled)

12. (Previously Presented) The method of claim 1, further comprising:

re-defining the area automatically if the content of the selected layer of the electronic

artwork is edited, the re-defined area conforming to a new perimeter boundary of the one or more non-transparent regions.

13. (Previously Presented) In a graphics application that supports dynamic content in layers, the method of claim 3, further comprising:

calculating any dynamic content for the selected layer when the layer is composited; and

using the calculated dynamic content to calculate the perimeter boundary and define the area.

14. (Cancelled)

15. (Previously Presented) The method of claim 1, wherein:

the image data in the selected layer has two or more non-contiguous non-transparent regions; and

the two or more non-contiguous non-transparent regions in combination are used to calculate the perimeter boundary.

16. (Previously Presented) The method of claim 15, further comprising:

generating multiple image maps from the non-transparent regions.

17-19. (Cancelled)

20. (Previously Presented) The computer program of claim 5, further comprising instructions for causing a computer to:

associate the area and the action with the selected layer as a property of the selected layer.

21. (Previously Presented) The computer program of claim 20, further comprising instructions for causing a computer to:

re-define the area automatically if the content of the selected layer of the electronic

artwork is edited, the re-defined area conforming to a new perimeter boundary of the one or more non-transparent regions.

22. (Previously Presented) The computer program of claim 7, further comprising instructions for causing a computer to:

calculate any dynamic content for the selected layer when the layer is composited; and use the calculated dynamic content to calculate the boundary and define the area.

23. (Cancelled)

24. (Previously Presented) The computer program of claim 5, wherein:

the image data in the selected layer has two or more non-contiguous non-transparent regions; and

the two or more non-contiguous non-transparent regions in combination are used to calculate the perimeter boundary.

25. (Previously Presented) The computer program of claim 24, further comprising instructions for causing a computer to:

generate multiple image maps from the non-transparent regions.

26-27. (Cancelled)

28. (Previously Presented) The method of claim 1, wherein:

defining the area further comprises automatically fitting a shape to the perimeter boundary, wherein the shape defines the area.

29. (Previously Presented) The method of claim 3, further comprising:

outputting the final image as an image file; and

outputting an HTML file including an image map for the area and a URL for the action.

30. (Previously Presented) The computer program of claim 5, wherein the action is a URL (Uniform Resource Locator).

31. (Previously Presented) The method of claim 28, further comprising:

receiving user input selecting the shape.

32. (Previously Presented) The method of claim 28, wherein the shape is a circle.

33. (Previously Presented) The method of claim 28, wherein the shape is a rectangle.

34. (Previously Presented) The method of claim 28, wherein the shape is a polygon.

35. (Previously Presented) The method of claim 1, wherein the perimeter boundary is for the one or more non-transparent regions in combination.

36. (Previously Presented) The method of claim 1, wherein there are one or more holes within the one or more non-transparent regions; and

wherein the holes are ignored in calculating the perimeter boundary.

37. (Previously Presented) The method of claim 1, wherein there are one or more holes within the one or more non-transparent regions; and

each non-transparent region that has a hole is separated into separate non-transparent regions that do not contain holes; and

the perimeter boundary is calculated from the separate hole-free non-transparent regions.

38. (Previously Presented) The computer program of claim 6, further comprising instructions to:

receive user input selecting the shape.

39. (Previously Presented) The method of claim 38, wherein the shape is a circle.

40. (Previously Presented) The method of claim 38, wherein the shape is a rectangle.

41. (Previously Presented) The method of claim 38, wherein the shape is a polygon.

42. (Previously Presented) The computer program of claim 5, wherein the boundary is for the one or more non-transparent regions in combination.

43. (Previously Presented) The computer program of claim 5, wherein there are one or more holes within the one or more non-transparent regions; and

wherein the holes are ignored in calculating the perimeter boundary.

44. (Previously Presented) The computer program of claim 5, wherein there are one or more holes within the one or more non-transparent regions; and

each non-transparent region that has a hole is separated into separate non-transparent regions that do not contain holes; and

the perimeter boundary is calculated from the separate hole-free non-transparent regions.

45-46. (Cancelled)